Distributed Processing and Energy Saving Techniques in Mobile Crowd Sensing

Enrique V. Carrera

Department of Electrical Engineering

Ecuadorian Armed Forces University

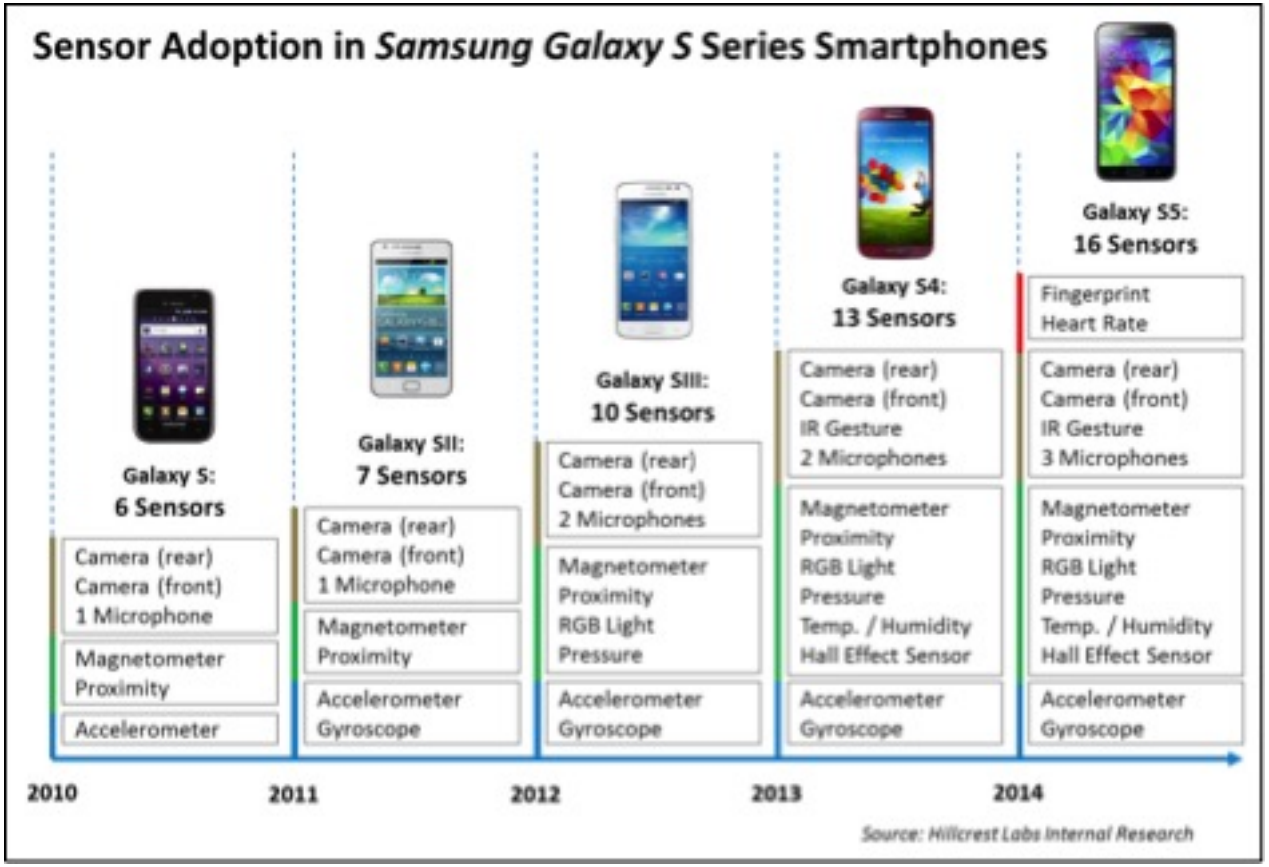
September 1st, 2016
Grenoble, France

Overview

Mobile Crowd Sensing

- Successful society and city management relies on efficient monitoring of urban and community dynamics for decision and policy making
- However, commercial sensor network techniques have never been successfully deployed in the real world due to several reasons, such as
  - High installation cost
  - Insufficient spatial coverage
- MCS is a large-scale sensing paradigm based on the power of user-companioned devices, including mobile phones, smart vehicles, wearable devices, and so on [1]

Mobile Crowd Sensing

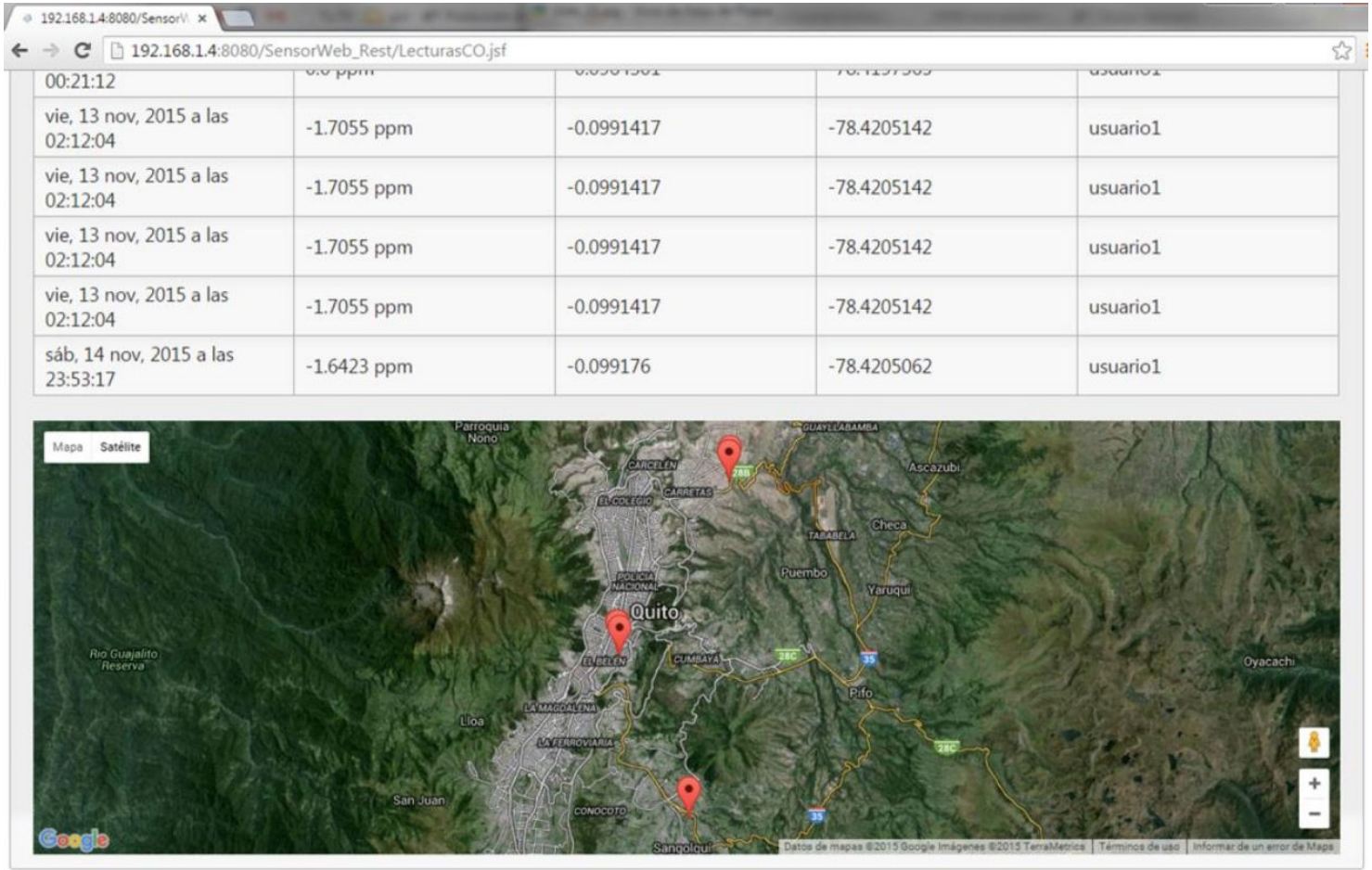


Typical smartphone sensors

Projects

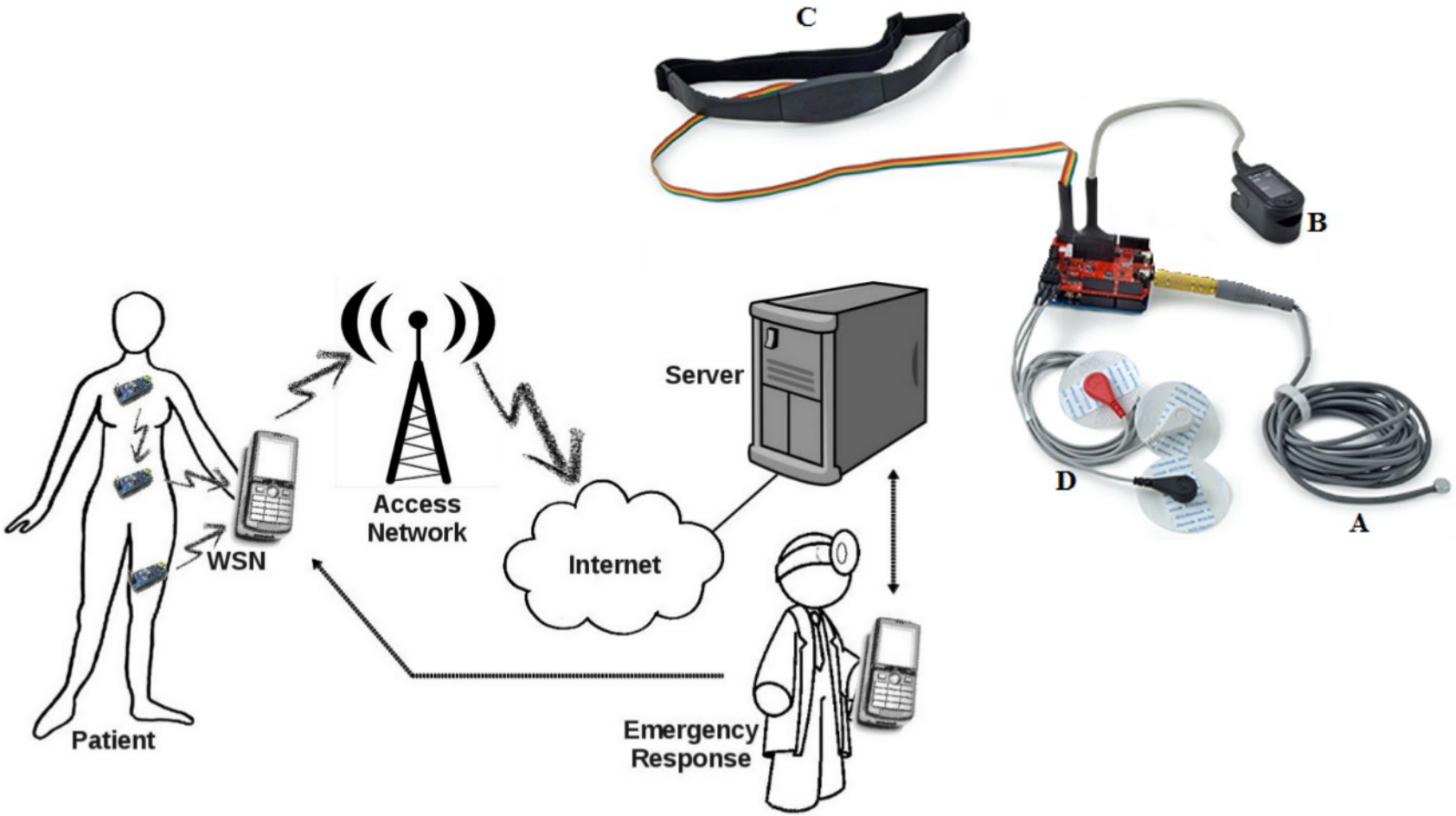
- The information must be aggregated in the cloud for large-scale sensing and community intelligence mining
- Previous projects:
  - Real-time environmental monitoring using smartphone apps
  - Healthcare monitoring of elderly people [2]
- Current projects:
  - Public safety monitoring through an Integrated Security Service
  - Earthquake monitoring based on smartphone sensors [3]

Projects



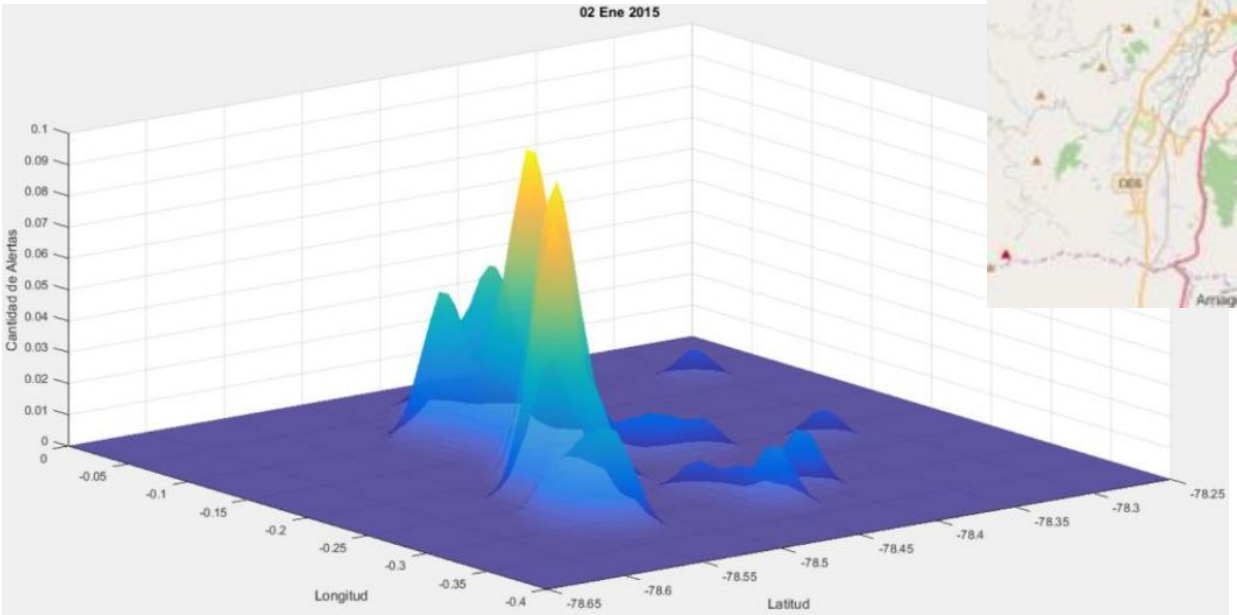
Environment monitoring [4]

Projects



Healthcare monitoring [5]

Projects



Public safety monitoring

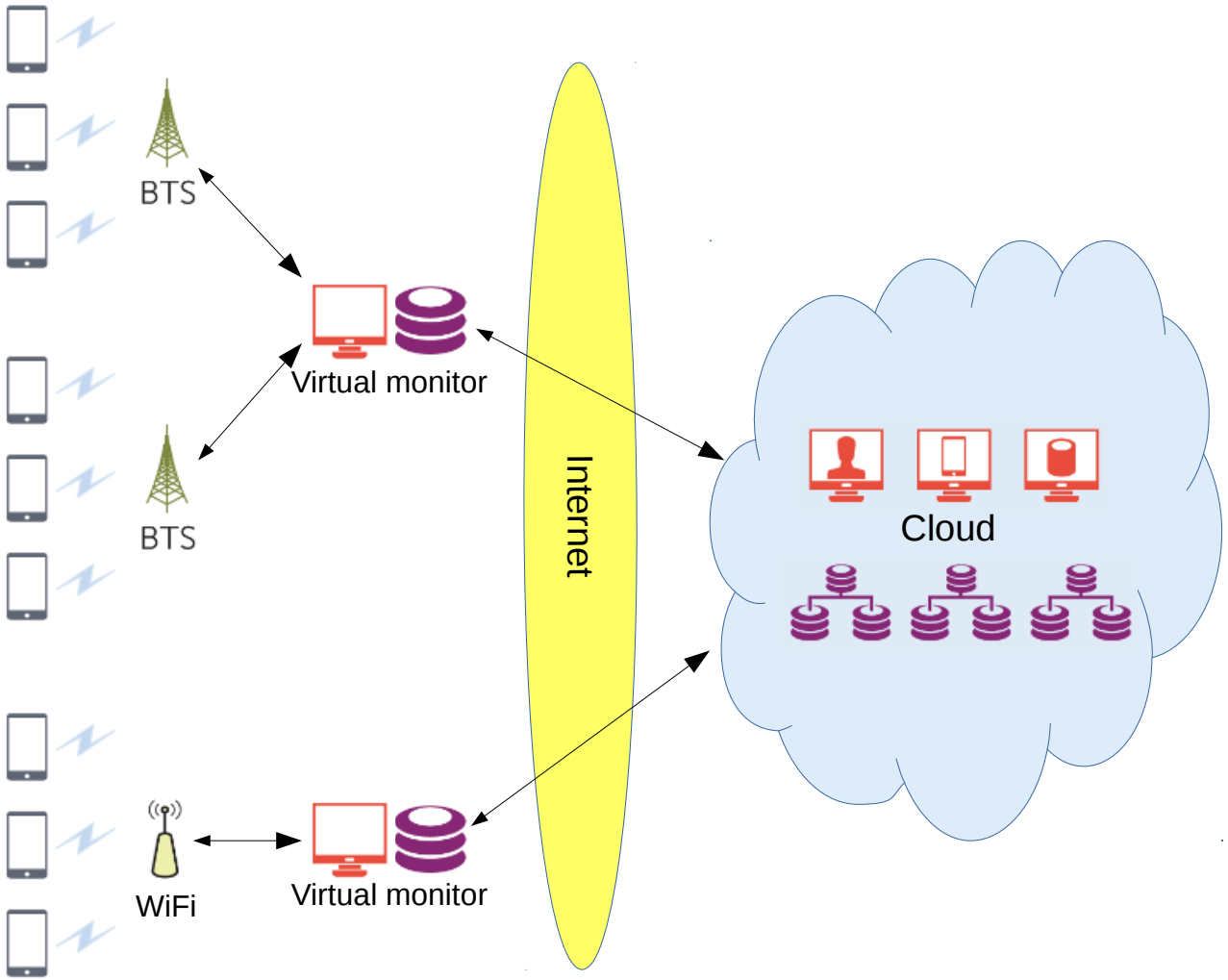
Research Gap

- Deploying MCS applications in large-scale environments is not a trivial task [6]
  - Heterogeneity of sensing hardware and mobile platforms
  - Increasing network bandwidth demands of emerging crowd sensing apps (*i.e.*, high data rate sensors)
  - Real-time processing is challenging because of high latencies
  - Participating users are exposed to a significant drain on limited mobile battery resources
- Proposed strategies produce [7]:
  - High user engagement
  - Poor data quality
  - Excessive energy consumption

Distributed Processing

- The dominant approach of aggregating all data to a single data center inflates the timeline of analytics [8]
- In fact, MCS normally relies on an Internet-scale searchable repository (*i.e.*, centralized analytics)
- Our proposal includes a *hierarchical distributed architecture* in order to:
  - Generate very low and predictable latencies
  - Solve scalability issues
- Then, the challenge is running queries over geo-distributed inputs by optimizing placement of both data and tasks of queries [9]

Distributed Processing



Hierarchical distributed architecture

Energy Saving

- In other words, we are looking for a *self-adaptive edge analytics* by pushing processing to the edge [10]
- However, mobile devices operates on a finite supply of energy contained in its battery
- In order to reduce energy costs without sacrificing precision of data, our strategy is combining:
  - Piggybacking based on smartphone app opportunities
  - Lightweight compression of data
  - Simple anomaly and outlier detection [11]

Methodology

- Distributed processing simulation:
  - Virtual smartphones sending fake accelerometer measurements
  - Virtual monitors receive sensor data and apply some analytics
  - Centralized server keeps summary of recorded activity

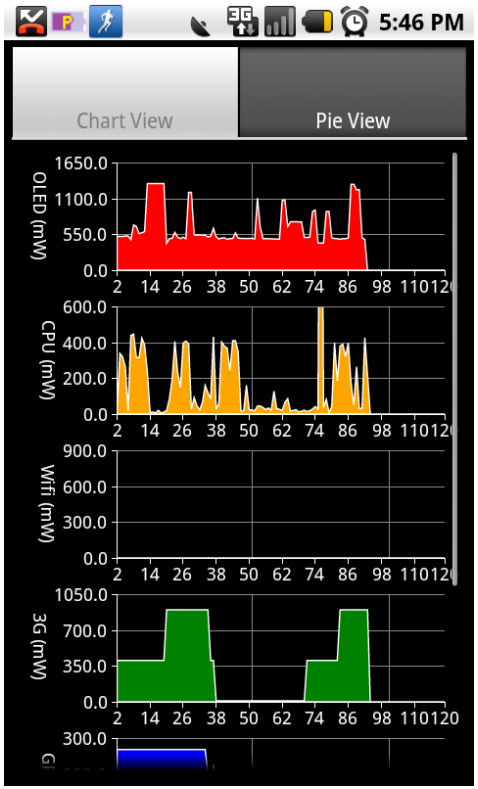
    (HPC cluster: Linux, 320 GB, 730 cores – up to 4096 nodes)
- Smartphone energy consumption characterization [12]:
  - Hardware (NI myDAQ)
  - Software (Android apps)

Evaluation

- Distributed processing simulation:
  - Throughput supported by a single virtual monitor
  - Experienced latency among tiers
  - Performance of distributed queries
- Smartphone energy consumption:
  - Energy consumption of activated sensors
  - Energy consumption in function of sensor data rate
  - Energy consumption of data compression algorithms
  - Energy consumption of simple anomaly detection algorithms

Evaluation

NI myDAQ



PowerTutor

Conclusions

- We have motivated MCS — a cross-space, heterogeneous crowdsourced sensing paradigm for large-scale sensing and computing
- MCS will foment and enhance numerous application areas, such as environment monitoring, intelligent transportation, urban sensing, mobile social recommendation, and so on
- However, the deployment of MCS applications in large-scale environments is a challenging task
- Then, we are proposing a hierarchical distributed architecture where processing is pushed to the edge without increasing energy consumption of battery-operated devices

Conclusions

- For evaluating our proposal, we are implementing a simulation of the distributed architecture plus actual power consumption measurements in smartphones
- We expect to get our first results once the simulated platform is working at the end of September 2016
- After that, we are going to start working in enhancements to our original proposal
- There are many research opportunities related to distributed processing and energy consumption issues in the field of MCS

References

1. H. Ma *et al.* (2014) "Opportunities in mobile crowd sensing." IEEE Comms Mag.
2. P. Guano *et al.* (2015) "A portable electronic system for health monitoring of elderly people." IEEE Colcom
3. R. Lara *et al.* (2016) "Automatic recognition of long period events from volcano tectonic earthquakes at Cotopaxi volcano." IEEE TGRS
4. S. Perez, E. V. Carrera. (2015) "Time synchronization in Arduino-based wireless sensor networks." IEEE LATrans
5. E. V. Carrera *et al.* (2012) "ECG signal monitoring using networked mobile devices." IEEE Andescon
6. Y. Xiao *et al.* (2013) "Lowering the barriers to large-scale mobile crowdsensing." ACM WMCSA

References

❼ N. Lane *et al.* (2013) "Piggyback CrowdSensing (PCS): energy efficient crowdsourcing of mobile sensor data by exploiting smartphone app opportunities." ACM CENSS

❽ A. Giordano *et al.* (2016) "Smart agents and Fog computing for smart city applications." Sringer ICSC

❾ Q. Pu *et al.* (2015) "Low latency geo-distributed data analytics." ACM Computer Communication Review

❿ M. Satyanarayanan *et al.* (2015) "Edge analytics in the internet of things." IEEE Pervasive

⓫ S. Kartakis, J. McCann. (2014) "Real-time edge analytics for cyber physical systems using compression rates." Usenix ICAC

⓬ M. Hoque *et al.* (2016) "Modeling, profiling, and debugging the energy consumption of mobile devices." ACM Computing Surveys